Amendments to the Specification

Please replace the Table 1 on pages 7 and 8 with the following table:

Step	Description	When executed
Master Resolution	Elect 1 Master CPU	Whenever the topology changes
Topology Discovery	Elected Master determines topology and assigns chip IDs/MAC Addresses to all VCPUs of Slave devices	
Remote Register Read/Write	Master issues Read/Write for remote registers. VCPU of slave devices interprets the command, performs the operations and sends a reply back to the Master.	When required by the Master.
BPDU and special Multicasts	BPDU and special Multicasts are encapsulated by Slave VCPU along with a header and sent to the Master.	BPDU/Special Mutlicasts are received by the Slave
MAC Table synchronization	VCPU sends "Leamed" and "Aged" messages to Master CPU.	MAC Table in slave changes.
Interrupt Processing	VCPU sends Interrupt information to Master	Enabled interrupt is received by VCPU of slave device.
Monitoring	Packet to be monitored by remote device is encapsulated by slave device and sent to remote	Packet to be monitored is received by VCPU of slave device.

Please replace the second table on page 8 with the following table:

SMA[0]	SMA[1]	SMA[2]	SMA[3]	SMA[4]	SMA[5]	SA[0]	SA[1]
SA[2]	SA[3]	SA[4]	SA[5]	TYPE[0]	TYPE[1]	Dest chip ID/Src chip ID	OPCODE =SetID
MsgID[0]	MsgID[1]	Rsv[0]	Rsv[1]	PAD	PAD	PAD	PAD
PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD
PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD
PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD
PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD
CRC[0]	CRC[1]	CRC[2]	CRC[3]				

Please replace the first table on page 10 with the following table:

SMA[0]	SMA[1]	SMA[2]	SMA[3]	SMA[4]	SMA[5]	SA[0]	SA[1]
SA[2]	SA[3]	SA[4]	SA[5]	TYPE[0]	TYPE[1]	Dest chip ID/Src chip ID	OPCODE =Read/Write
MsgID[0]	MsgID[1]	Rsv[0]	Rsv[1]	No. Dwords	Poll/Status	Rsv	rsv
Addr[0]	Addr[1]	Addr[2]	Addr[3]	PAD/ Data[0]	PAD/ Data[1]	PAD/ Data[2]	PAD/ Data[3]
...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...
...	...	...	...	...	...	PAD/ Data[26]	PAD/ Data[27]
CRC[0]	CRC[1]	CRC[2]	CRC[3]				

Please replace the second table on page 10 which continues to page 11 with the following table:

SMA[0]	SMA[1]	SMA[2]	SMA[3]	SMA[4]	SMA[5]	SA[0]	SA[1]
SA[2]	SA[3]	SA[4]	SA[5]	TYPE[0]	TYPE[1]	Dest chip ID/Src chip ID	OPCODE =Read/Write
MsgID[0]	MsgID[1]	Rsv[0]	Rsv[1]	No. Dwords	rsv	rsv	Rsv
Byte[0]	Byte[1]	…	…	…	…	…	…
…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…
…	…	…	…	…	…	Byte[30]	Byte[31]
CRC[0]	CRC[1]	CRC[2]	CRC[3]				

Please replace the first complete table on page 11 with the following table:

SMA[0]	SMA[1]	SMA[2]	SMA[3]	SMA[4]	SMA[5]	SA[0]	SA[1]
SA[2]	SA[3]	SA[4]	SA[5]	TYPE[0]	TYPE[1]	Dest chip ID/Src chip ID	OPCODE =Read/Write
MsgID[0]	MsgID[1]	Rsv[0]	Rsv[1]	No. Dwords	Poll/Status	rsv	rsv
Addr[0]	Addr[1]	Addr[2]	Addr[3]	Data[0]	Data[1]	…	…
…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…
…	…	…	…	…	…	Data[26]	Data[27]
CRC[0]	CRC[1]	CRC[2]	CRC[3]				

Please replace the second complete table on page 11 with the following table:

SMA[0]	SMA[1]	SMA[2]	SMA[3]	SMA[4]	SMA[5]	SA[0]	SA[1]
SA[2]	SA[3]	SA[4]	SA[5]	TYPE[0]	TYPE[1]	Dest chip ID/Src chip ID	OPCODE =ENCAP
MsgID[0]	MsgID[1]	Rsv	Rsv	PH[0]	PH[1]	PH[2]	PH[3]
PH[4]	PH[5]	PH[6]	PH[7]	EncPkt[0]	EncPkt[1]	EncPkt[2]	EncPkt[3]
EncPkt[4]	...	...	...	...	...	EncPkt[n-1]	EncPkt[n]
CRC[0]	CRC[1]	CRC[2]	CRC[3]				

Please replace the third complete table on page 11 with the following table:

Spid(5:0)			In_tagged	RuleID(9:0)
Rsv	Crcerr	Pkt_len(13:0)		
I_snapped	Vlan_id(11:0)			Pri(2:0)
Rsv(15:0)				

Please replace the first table on page 12 with the following table:

SMA[0]	SMA[1]	SMA[2]	SMA[3]	SMA[4]	SMA[5]	SA[0]	SA[1]
SA[2]	SA[3]	SA[4]	SA[5]	TYPE[0]	TYPE[1]	Dest chip ID/Src chip ID	OPCODE =ENCAP return
MsgID[0]	MsgID[1]	Rsv[0]	Rsv[1]	Dest port	Rsv	rsv	rsv
Rsv	Rsv	Rsv	Rsv	EncPkt[0]	EncPkt[1]	EncPkt[2]	EncPkt[3]
EncPkt[4]	...	...	...	...	...	EncPkt[n-1]	EncPkt[n]
CRC[0]	CRC[1]	CRC[2]	CRC[3]				

Please replace the second table on page 12 with the following table:

SMA[0]	SMA[1]	SMA[2]	SMA[3]	SMA[4]	SMA[5]	SA[0]	SA[1]
SA[2]	SA[3]	SA[4]	SA[5]	TYPE[0]	TYPE[1]	Dest chip ID/Src chip ID	OPCODE =Learned
MsgID[0]	MsgID[1]	Rsv[0]	Rsv[1]	MA[0]	MA[1]	MA[2]	MA[3]
MA[4]	MA[5]	SPID	PAD	PAD	PAD	PAD	PAD
PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD
PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD
CRC[0]	CRC[1]	CRC[2]	CRC[3]				

Please replace the third table on page 12 which continues to page 13 with the following table:

SMA[0]	SMA[1]	SMA[2]	SMA[3]	SMA[4]	SMA[5]	SA[0]	SA[1]
SA[2]	SA[3]	SA[4]	SA[5]	TYPE[0]	TYPE[1]	Dest chip ID/Src chip ID	OPCODE =Aged
MsgID[0]	MsgID[1]	Rsv[0]	Rsv[1]	MA[0]	MA[1]	MA[2]	MA[3]
MA[4]	MA[5]	PAD	PAD	PAD	PAD	PAD	PAD
PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD
PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD
CRC[0]	CRC[1]	CRC[2]	CRC[3]				

Please replace the first complete table on page 13 with the following table:

SMA[0]	SMA[1]	SMA[2]	SMA[3]	SMA[4]	SMA[5]	SA[0]	SA[1]
SA[2]	SA[3]	SA[4]	SA[5]	TYPE[0]	TYPE[1]	Dest chip ID/Src chip ID	OPCODE =Interrupt
MsgID[0]	MsgID[1]	Rsv[0]	Rsv[1]	IntStatus Reg[0]	IntStatus Reg[1]	IntStatus Reg[2]	IntStatus Reg[3]
PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD
PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD
PAD	PAD	PAD	PAD	PAD	PAD	PAD	PAD
CRC[0]	CRC[1]	CRC[2]	CRC[3]				

Please replace the second table on page 13 which continues to page 14 with the following table:

Opcode Name	**Message direction**	**Explanation**	**Code**
MasterResolution	Master → Master	Needs to occur if two stacks are connected together	0×00
ENCAPforward	Slave → Master	BPDU to Master CPU	0×01
ENCAPretum	Master → Slave	Master CPU sends BPDU for remote port	0×02
Read	Master → Slave	Master CPU issues read request for Slave	0×03
Write	Master → Slave	Master CPU issues write request for Slave	0×04
ReadAck	Slave → Master	Slave VCPU returns data.	0×05
WriteAck	Slave → Master	Slave VCPU issues write Acknowledge.	0×06
Error	Slave → Master	Error occurred while processing Msg with given ID.	0×07
SetID	Master → Slave	Master CPU requests first Slave with no Chip to assign an ID to itself.	0×08
SetIDAck	Slave → Master	Slave to Master.	0×09
ResetID	Master → Slave	Master CPU requests Slave to deassign Chip ID.	0×0A
ResetIDAck	Master → Slave	Slave to Master.	0×0B
Interrupt	Slave → Master	Slave sends interrupt register to Master.	0×0C
Learned	Slave → Master	Slave sends Learned message to CPU.	0×0D
Aged	Slave → Master	Slave sends Learned message to CPU.	0×0E

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claim 1-4 (canceled)

Claim 5 (currently amended). A network of data switches, each data switch having a plurality of ports adapted for receiving and transmitting packets and arranged for transferring data packets internally between the ports of the data switches according to address information in the packets, the data switches being connected as an array, the array formed by connections between ports of pairs of the switches, the network of data switches including a master switch and other data switches, the master switch configured to issue commands to the other data switches, the commands in the form of control data packets, the other data switches comprising slave data switches configured to recognize the control data packets and to operate based on the commands contained within the control data packets, wherein a first slave data switch is further operable to pass a control data packet from the first slave data switch to a second slave data switch if the first slave data switch determines that the control data packet is not intended to cause the command to be carried out at the first slave data switch.

Claim 6 (previously presented). The network of data switches according to claim 5, wherein the master data switch is further operable to determine a topology of the network of data switches.

Claim 7 (currently amended). The network of data switches according to claim 5, wherein each slave data switch is further configured [to] implement a command within a control data packet if the slave data switch determines that the control data packet is intended to cause the command to be carried out at the slave data switch.

Claim 8 (canceled).

Claim 9 (currently amended). Method of operating a plurality of data switches, each data switch having a plurality of ports adapted for receiving and transmitting packets and arranged for transferring data packets internally between [the] ports ~~of others~~ of the ~~plurality of~~ data switches according to address information in the data packets, the method comprising:

employing at least one port of a master data switch of the plurality of data switches to issue command packets to slave data switches of the plurality of switches;

employing at least one port of each of the slave data switches to receive the command packets;

recognizing within the slave data switches the command packets and implementing commands specified in the command packets; and

passing a command packet from a first slave data switch to a second slave data switch if the first slave data switch determines that the command packet is not intended to cause a command to be carried out at the first slave data switch.

Claim 10 (currently amended). The method according to claim 9, wherein the recognizing step further comprises determining ~~at a~~ at the first slave data switch ~~whether a~~ whether the command packet transmitted to the first slave data switch is intended to ~~cause a~~ cause the command within the command packet to be carried out at the first slave data switch.

Claim 11 (previously presented). The method of claim 10 further comprising implementing the command at the first slave data switch if the first slave data switch determines that the command packet is intended to cause the command to be carried out at the first slave data switch.

Claim 12 (canceled).

Claim 13 (currently amended). A method according to ~~claim 10~~ claim 11 further comprising:

determining at the master data switch a topology of the network of data switches.

Claim 14 (previously presented). The method according to claim 13, further comprising assigning IDs to the slave data switches, said IDs included in subsequent packets passing between the switches within the network of data switches.

Claim 15 (previously presented). A method according to claim 9, further comprising:

determining, under the control of the master data switch, a topology of the network of data switches.

Claim 16 (previously presented). The method according to claim 15, further comprising assigning IDs to the slave data switches, said IDs included in subsequent packets passing between the switches within the network of data switches.